Hadoop HA HDFS集群搭建

# 1、环境准备

1、VMware Workstation Pro11（版本自己选）

2、centos6.6(版本自己选)

3、zookeeper3.4.8

4、hadoop2.5.2

5、apache-hive-0.13.1

6、jdk-7u79-linux-x64.tar



# 2、搭建linux服务器四台

过程略

编辑/etc/hosts

192.xx.xx.1 node1 node1

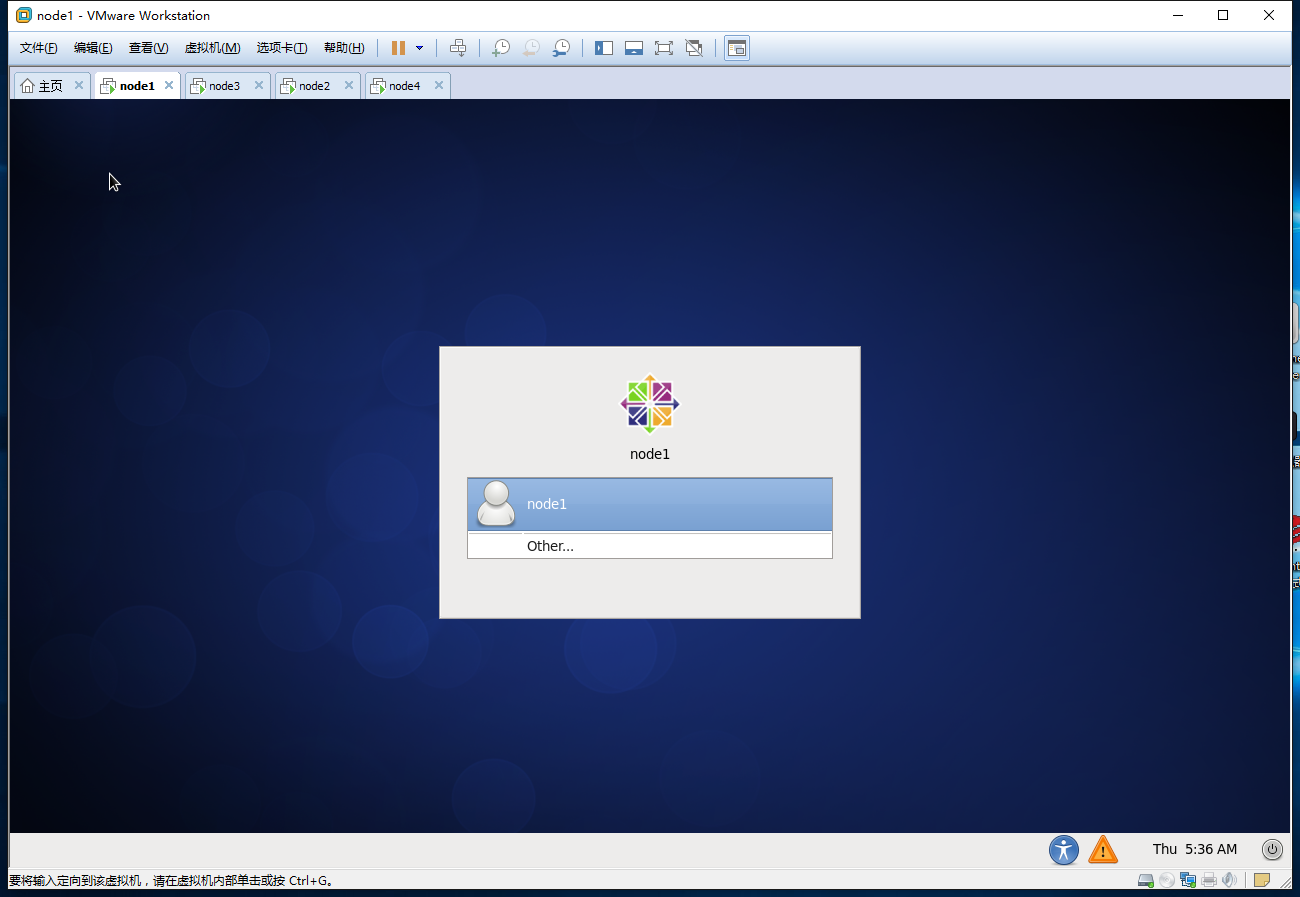
192.xx.xx.1 node2 node2

192.xx.xx.1 node3 node3

192.xx.xx.1 node3 node4

设置hostname

运行hostname nodeX



免密码登录，让node1能够密码登录到node2、node3、node4

$ ssh-keygen -t dsa -P '' -f ~/.ssh/id\_dsa

$ cat ~/.ssh/id\_dsa.pub >> ~/.ssh/authorized\_keys

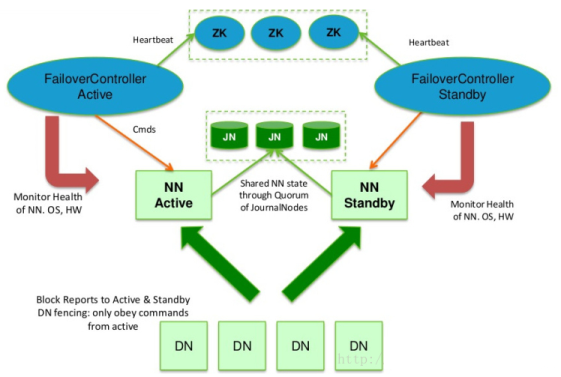
注意将id\_dsa.pub拷贝到node2、node3、node4上，然后再使用cat命令追加到本机的~/.ssh/authorized\_keys，否则容易出现编码问题导致身份认证失败。

作用：hadoop启动时，能够ssh到其他机器执行命令

关闭node1 node2 node3 node4的防火墙设置

Service iptables stop

# 3、HA HDFS原理



经过分析：

节点数：HA HDFS节点13个+ MR节点1+3个

至少需要3台服务器（本文是以四台服务器做集群）

节点分布：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | NN | DN | JN | ZKFC | ZK | RM | DM |
| Node1 | √ |  |  | √ | √ | √ |  |
| Node2 | √ | √ | √ | √ | √ |  | √ |
| Node3 |  | √ | √ |  | √ |  | √ |
| Node4 |  | √ | √ |  |  |  | √ |

# 4、JDK安装

略

配置环境变量 vi /etc/profile

export JAVA\_HOME=/usr/opt/jdk1.7.0\_79

export JRE\_HOME=${JAVA\_HOME}/jre

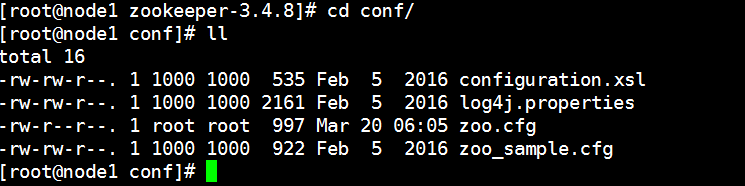
export CLASSPATH=.:${JAVA\_HOME}/lib:${JRE\_HOME}/lib

export PATH=${JAVA\_HOME}/bin:$PATH

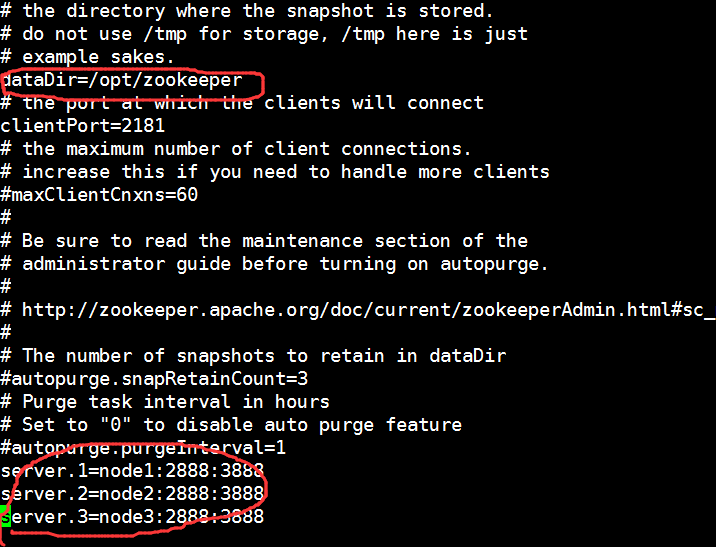
# 5、Zookeeper集群搭建

1、分别在node1、node2、node3上解压tar –zxvf zookeeper-3.4.8.tar.gz

2、拷贝配置文件cp zoo\_sample.cfg zoo.cfg



3、修改zookeeper的配置文件



4、在node1 node2 node3的/opt/zookeeper/下分别新建myid文件touch myid

文件的内容分别写入上图配置的服务器的编号，例如node1的myid内容1，node2的myid内容为2，node3的myid内容为3

5、zkServer.sh start分别启动node1 node2 node3的zk

6、jps查看是否存在QuorumPeerMain进程，也可以查看日志bin/zookeeper.out确保启动

7、使用zkCli.sh客户端连接zk（略）

# 6、HA HDFS搭建

以node1为例，拷贝安装包到/usr/opt目录下，解压安装包



hadoop2.5.2官网是32位的，在网上搜索一个编译打包好的64位的

解压 tar –zxvf hadoop-aboutyun-Linux64-2.5.2-.tar.gz

创建一个软连接 ln –sf /usr/opt/hadoop-2.5.2 /home/hadoop-2.5

下面是修改配置文件（可以查阅官网的配置过程）/home/hadoop-2.5/etc/hadoop/下

hdfs-site.xml

[root@node1 ~]# vi /home/hadoop-2.5/etc/hadoop/hdfs-site.xml

<!—hadoop对外服务的集群服务名-->

<property>

<name>dfs.nameservices</name>

<value>highpay</value>

</property>

<!—hadoop NN-->

<property>

<name>dfs.ha.namenodes.highpay</name>

<value>nn1,nn2</value>

</property>

<!—hadoop NN1 rpc通信地址-->

<property>

<name>dfs.namenode.rpc-address.highpay.nn1</name>

<value>node1:8020</value>

</property>

<!—hadoop NN2 rpc通信地址-->

<property>

<name>dfs.namenode.rpc-address.highpay.nn2</name>

<value>node2:8020</value>

</property>

<!—hadoop NN1 http通信地址-->

<property>

<name>dfs.namenode.http-address.highpay.nn1</name>

<value>node1:50070</value>

</property>

<!—hadoop NN2 http通信地址-->

<property>

<name>dfs.namenode.http-address.highpay.nn2</name>

<value>node2:50070</value>

</property>

<!—hadoop JN-->

<property>

<name>dfs.namenode.shared.edits.dir</name>

<value>qjournal://node2:8485;node3:8485;node4:8485/highpay</value>

</property>

<!—hadoop ZKFC-->

<property>

<name>dfs.client.failover.proxy.provider.highpay</name>

<value>org.apache.hadoop.hdfs.server.namenode.ha.ConfiguredFailoverProxyProvider</value>

</property>

<!—hadoop sshfence-->

<property>

<name>dfs.ha.fencing.methods</name>

<value>sshfence</value>

</property>

<property>

<name>dfs.ha.fencing.ssh.private-key-files</name>

<value>/root/.ssh/id\_rsa</value>

</property>

<!—hadoop hdfs HA -->

<property>

<name>dfs.ha.automatic-failover.enabled</name>

<value>true</value>

</property>

core-site.xml

<configuration>

<!—hadoop hdfs 临时文件，默认放在用户的临时目录，重启会导致数据丢失-->

<property>

<name>hadoop.tmp.dir</name>

<value>/home/hdfs</value>

</property>

<!—hadoop hdfs访问入口-->

<property>

<name>fs.defaultFS</name>

<value>hdfs://highpay</value>

</property>

<!—hadoop hdfs edits文件存放路径 -->

<property>

<name>dfs.journalnode.edits.dir</name>

<value>/usr/opt/hadoop2</value>

</property>

<!—zk集群配置-->

<property>

<name>ha.zookeeper.quorum</name>

<value>node1:2181,node2:2181,node3:2181</value>

</property>

</configuration>

创建在上面配置文件中指定的目录

mkdir /home/hdfs

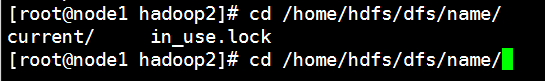
mkdir /usr/opt/hadoop2

如果此时启动./home/hadoop-2.5/sbin/start-hdf.sh，你认为可以启动hdfs吗？显然不能，原因很简单，还没有初始化NN等节点。

下面就是初始化节点

首先初始化NN，记住有2台NN

1. node1上初始化NN *hdfs namenode –format*



验证启动node1上的NN，到sbin目录下：

启动node1上的NN ./hadoop-daemon.sh start namenode

JPS查看是否Namenode进程，若失败查看日志

1. node2上NN怎么初始化呢？
2. 直接将node1上的元数据拷贝到node2上
3. 启动node1上NN，然后运行在node2上运行*hdfs namenode –bootstrapStandby*

同样验证下node2上NN是否可以正常启动

此时node1，node2上的NN都能正常启动，是否就代表整个HDFS集群可以正常启动呢？

关闭node1，node2上的NN ./hadoop-daemon.sh namenode stop

然后尝试启动/home/hadoop-2.5/sbin/start-hdf.sh，可以启动HA HDFS。

检查节点数13个

发现缺少2个节点，node1、node2上的ZCFC节点都没有启动。

查看日志vi ../logs/hadoop-root-zkfc-node1.log ，发现错误

Failed DFSZKFailoverController cannot initiating…..

因为ZCFC没有初始化导致

初始化ZKFC

在node1上执行./home/hadoop-2.5/bin/hdfs zkfc –formatZK

启动ZCFK，验证是否启动成功

./home/hadoop-2.5/sbin/hadoop-daemon.sh start zkfc

JPS查看java进程，发现node1,node2上的DFSZKFailoverController进程

关闭启动节点 ./stop-dfs.sh，准备启动HA HDFS

到此只是完成了在node1上的配置了HA HDFS配置过程，将node1的安装包以及配置文件全部拷贝到node2,node3,node4,并分别创建HDFS的相关目录保持与node1一致。

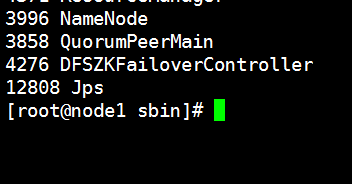
mkdir /home/hdfs

mkdir /usr/opt/hadoop2

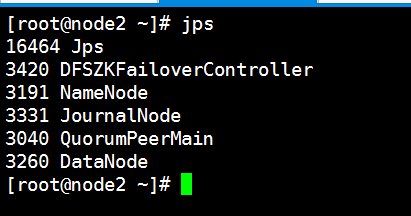
在node1上运行 ./home/hadoop-2.5/sbin/start-hdf.sh

JPS查看四台服务器的节点数

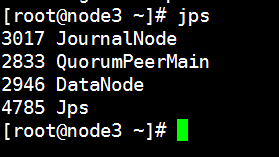
Node1



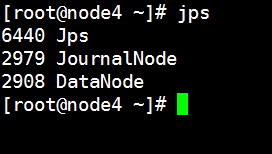
Node2



Node3



Node4



发现四台机器，13个节点，按照预期全部启动，可以通过下面的地址访问。

<http://node1:50070>

<http://node2:50070>

NN只有一个处于active，另一个处于standby

HA HDFS服务器集群搭建完毕